佐野研究室
Sano Laboratory

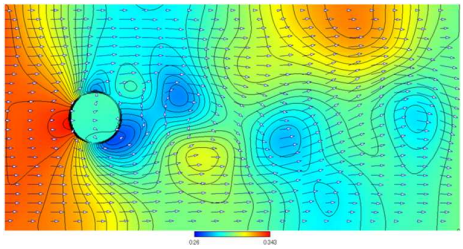
Scalability Analysis of Tightly-coupled FPGA-cluster for Lattice Boltzmann Computation

Yoshiaki Kono, Kentaro Sano, Satoru Yamamoto

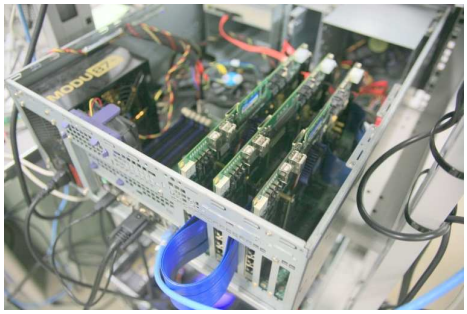
Graduate School of Information Sciences,
Tohoku University, Japan

Outline

- Introduction
- Lattice Boltzmann method (LBM)
- FPGA-cluster for LBM
- Performance model & analysis
- Conclusions

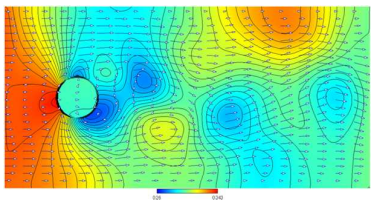


LBM computation



FPGA cluster

Introduction



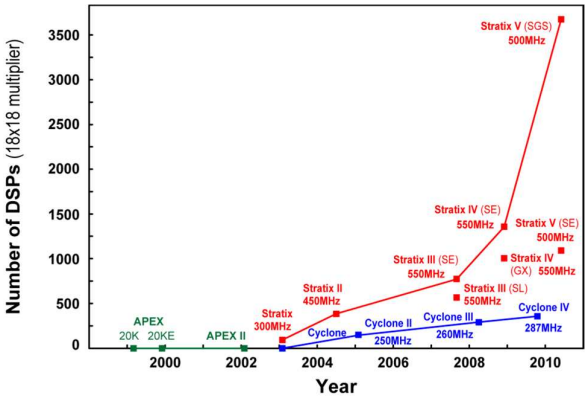
- **Lattice Boltzmann method (LBM)**
  : method to compute fluid dynamics
  - ✓ High parallelism
  - ✓ Low operational intensity (each op. requires many data)

For large-scale parallel computing….

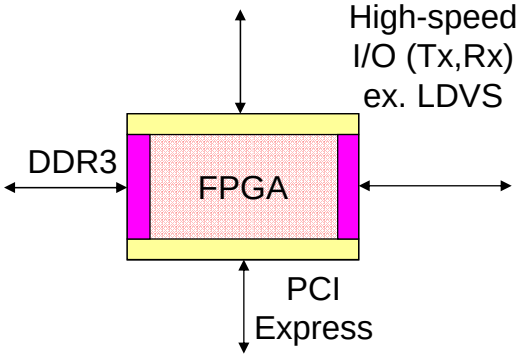
- **Today's micro-processors**
  - ✓ cannot provide peak-performance.
  - ✓ memory bandwidth is insufficient.
  - ✓ Limited scalability in large scale sys., conspicuous for strong scaling.
  - ✓ caused by imbalanced performance and bandwidth.
- **Custom-computing machine**
  - ✓ High utilization if we design appropriate HW
    (performance & bandwidth are balanced)

FPGA-based Custom Computing for HPC

- FPGAs have been getting larger and faster



High-end FPGAs now have performance of 1TFlops



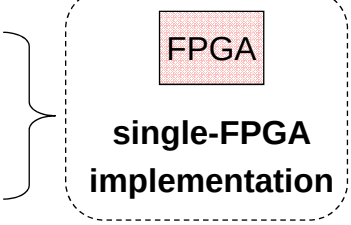
Higher bandwidth memories and chip-I/Os

Promising devices for custom HPC

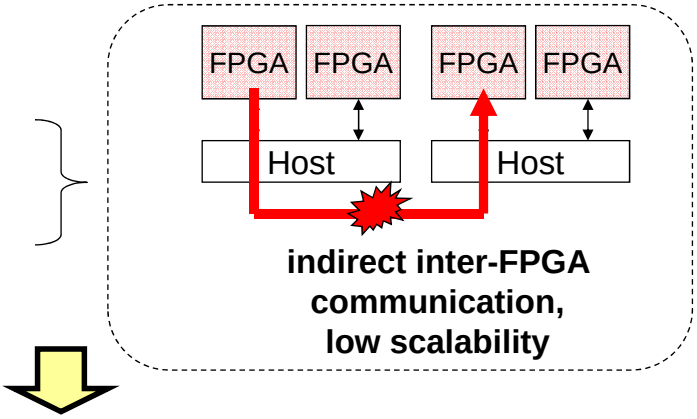
Related Work : Custom LBM Machines

[Sano2007]
FPGA-based streaming computation for LBM

[Murtaza2009]
Custom-computing machine for cellular-automata



[Murtaza2011]
LBM computation on
Maxwell (multi-FPGA system)

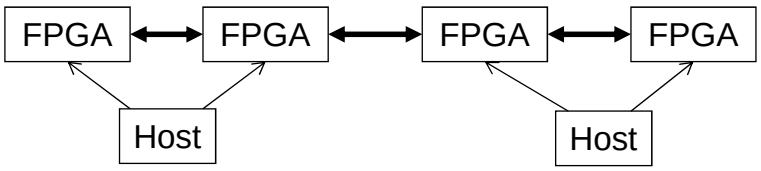


Tightly-coupled FPGA cluster with a dedicated network

Objectives of this Research

Strong-scalability analysis of tightly-coupled FPGA-cluster

- Direct connection between FPGAs beyond the node



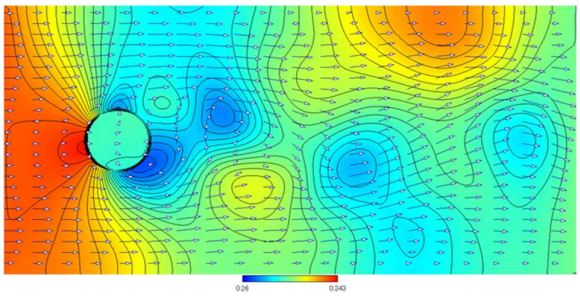
Architecture design for scalable LBM on FPGA-cluster

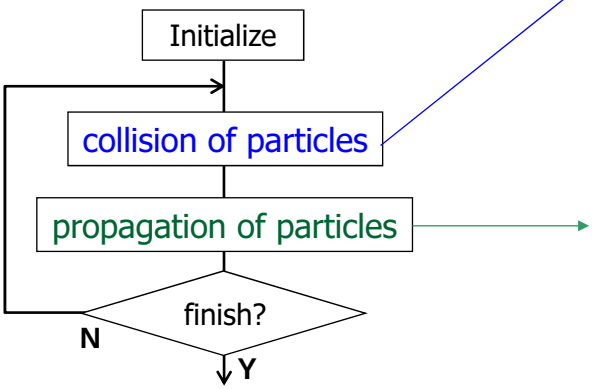
- Spatial- and Temporal- Parallelism

Contributions of this paper

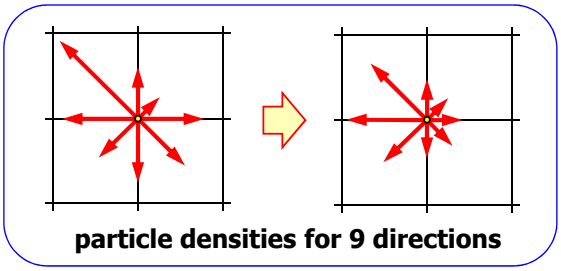
- ✓ Scalable-architecture design for LBM computation
- ✓ Sustained-performance model
- ✓ Analysis of strong-scalability

Lattice Boltzmann method : LBM

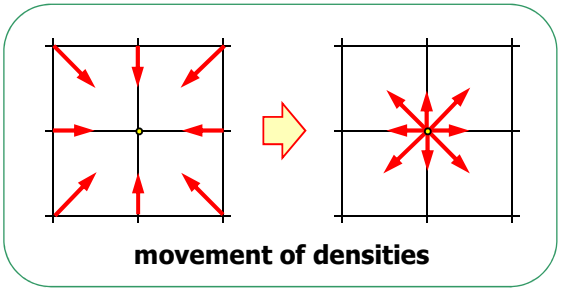




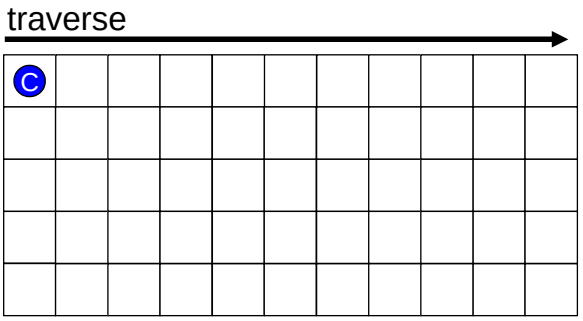
collision of particles

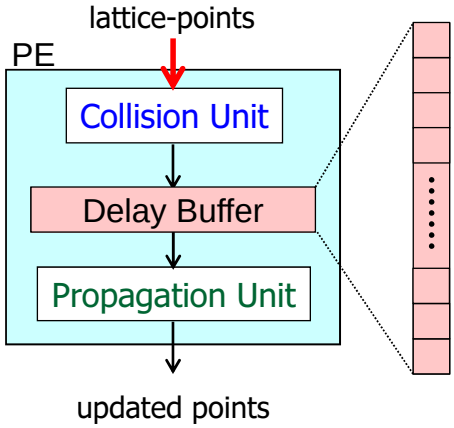


propagation of particles (gather)

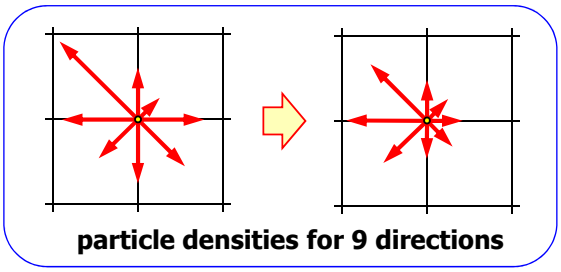


Processing Element for Stream Computation

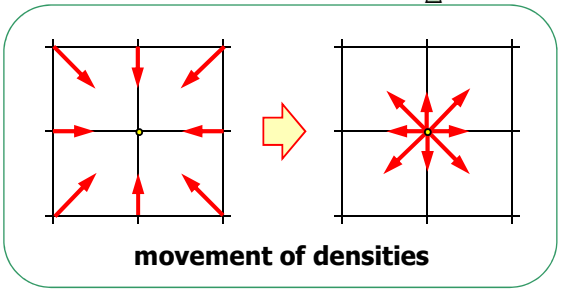




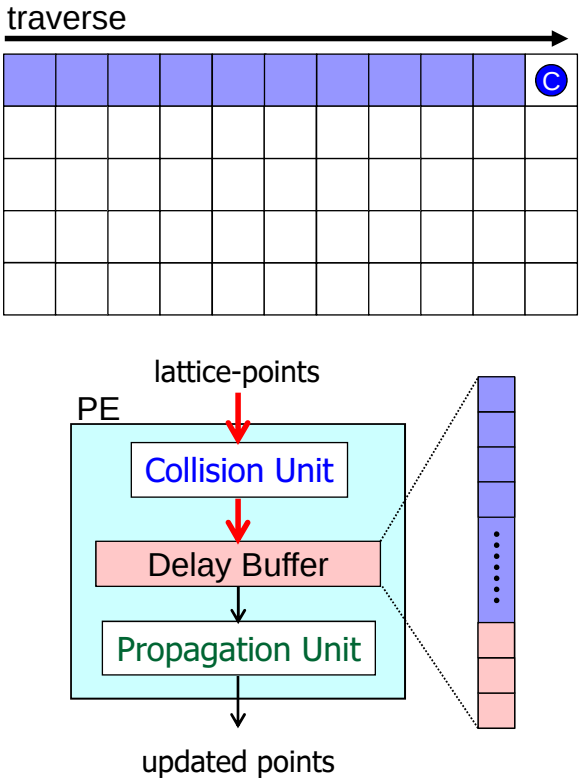
collision of particles C

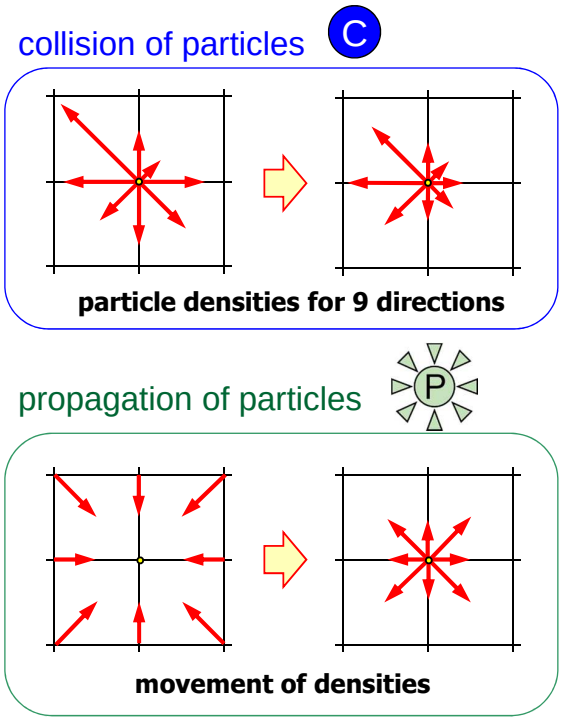


propagation of particles P

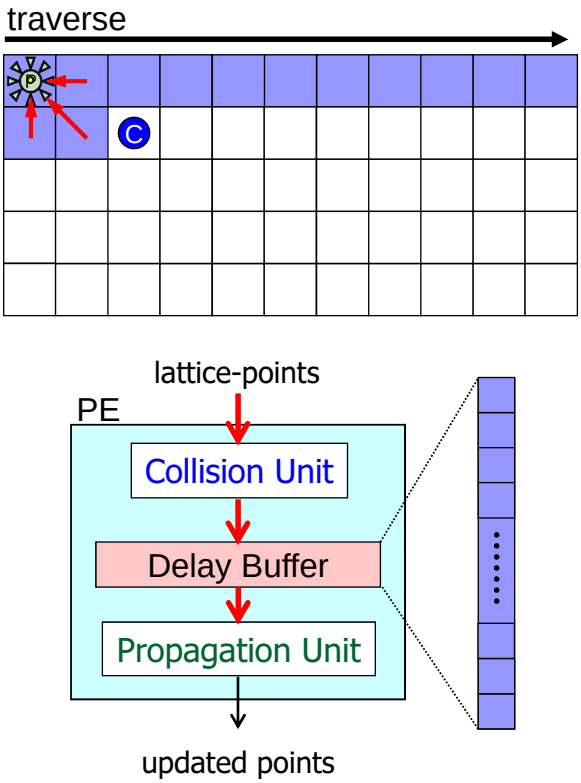


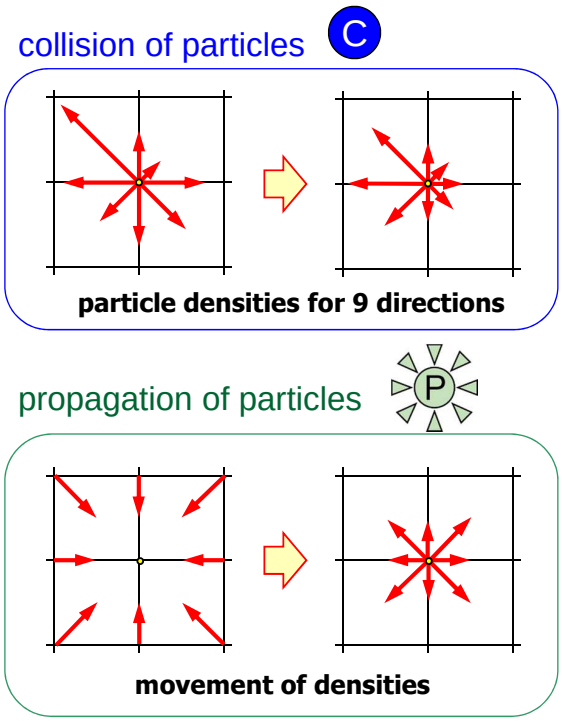
Processing Element for Stream Computation



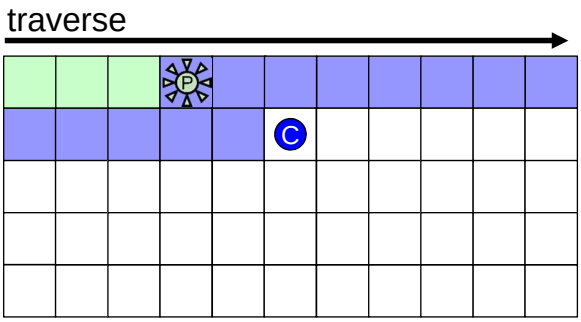


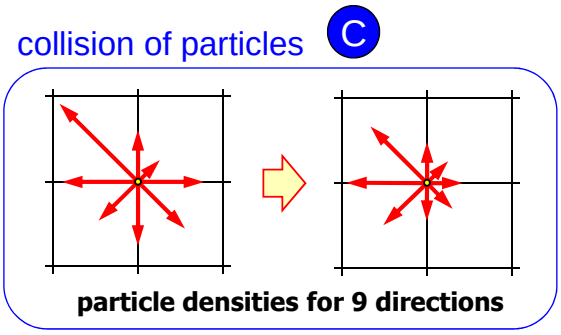
Processing Element for Stream Computation

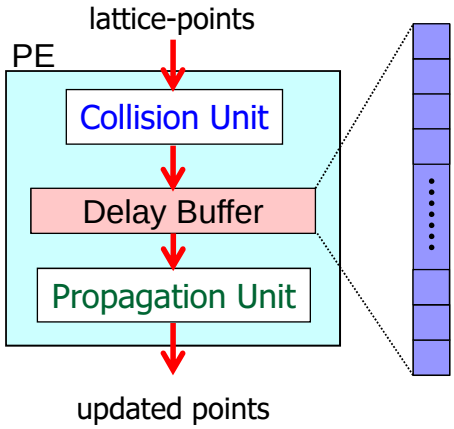


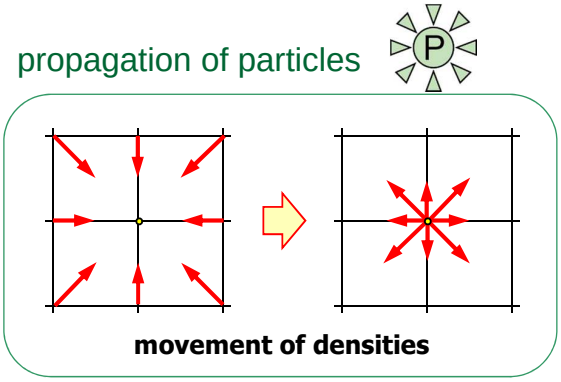


Processing Element for Stream Computation

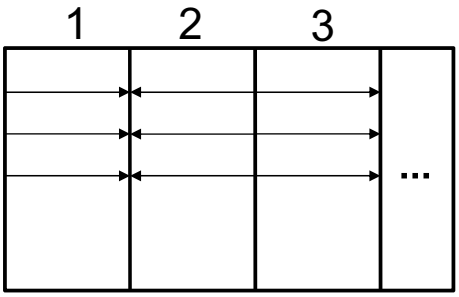


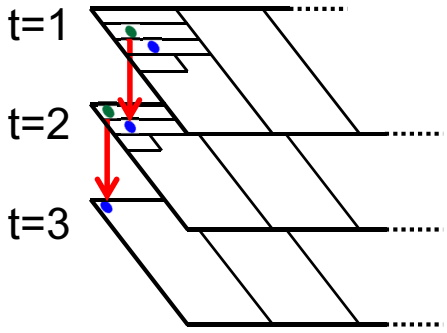






Parallelisms of Stream Computation



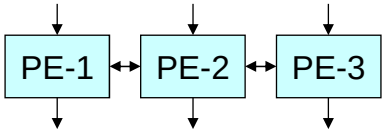


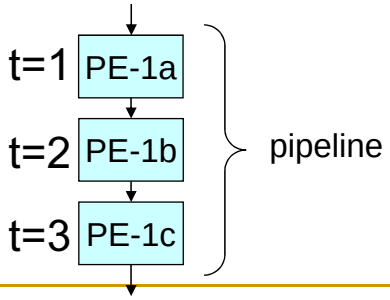
Spatial parallelism

- ✓ sub-lattices can be computed in parallel

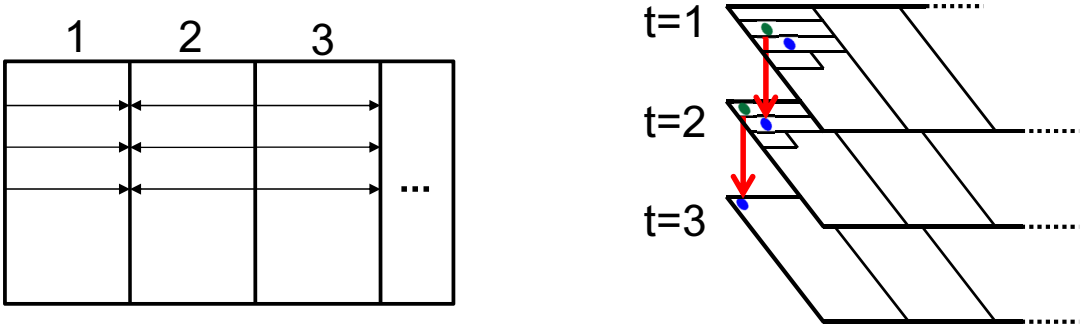
Temporal parallelism

- ✓ stream-computations can be pipelined along time-steps

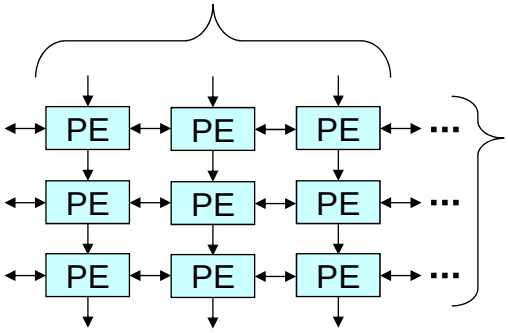




PE Array for Spatial and Temporal Parallelisms

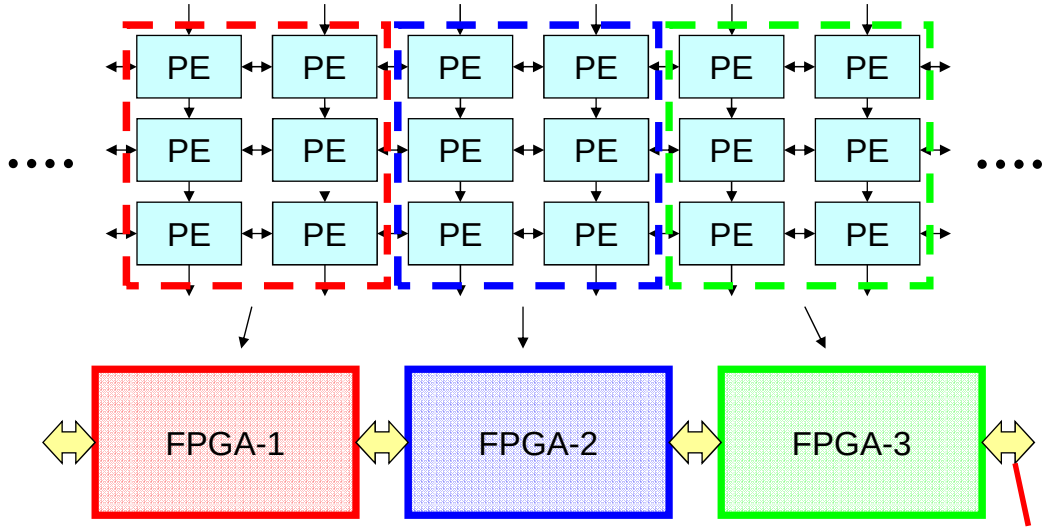


Spatial parallelism



Temporal parallelism

Large PE-Array on Tightly-Coupled FPGAs

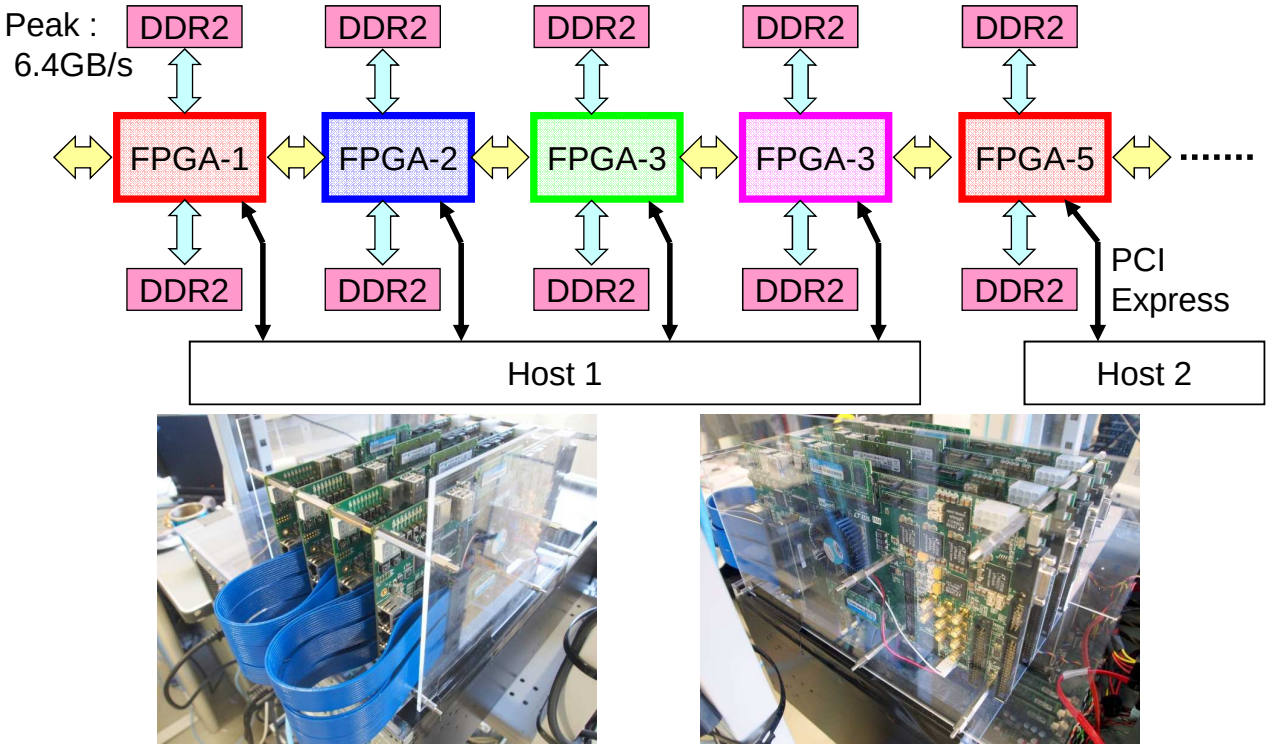


Tightly-coupled FPGA-cluster

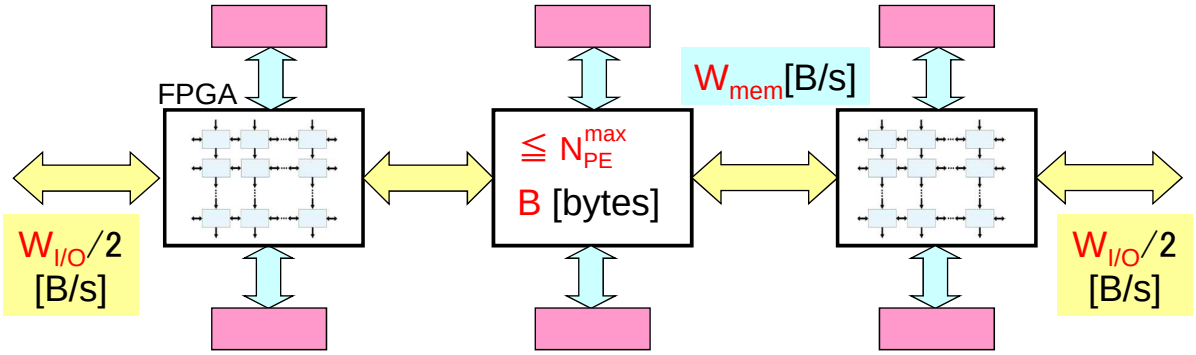
Accelerator-domain network (ADN)

- 1D-ring network for higher bandwidth between FPGAs
- 1D decomposition of a large PE-array

Prototype FPGA-Cluster



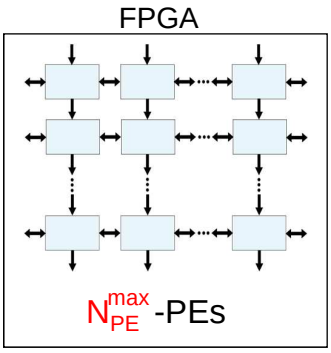
Parameters of FPGA-Cluster

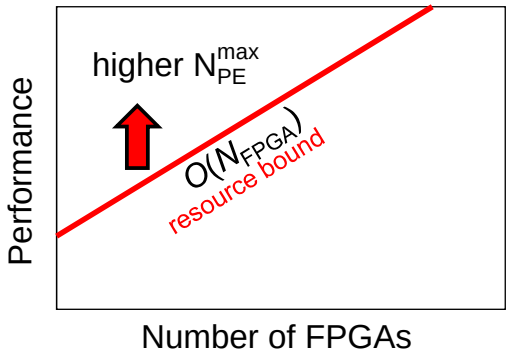


- W_{mem} : Memory bandwidth
- $W_{I/O}$: Network bandwidth
- B : BRAM size that can be used as buffers for propagation
- N_{PE}^{max} : Max num. of PEs that can be implemented on each FPGA

→ These Parameters influence Performance and Scalability

Performance Limited by Resources (# of PEs)





- Assuming 100% utilization,

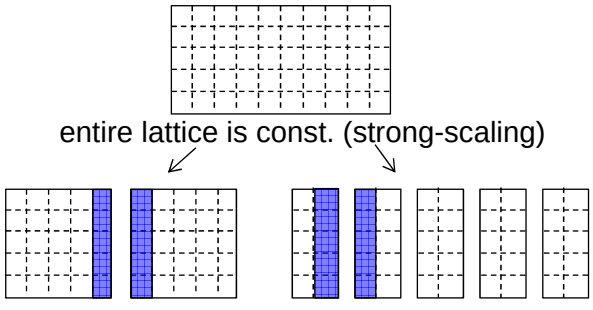
$$(\textit{Peak performance of each FPGA}) \propto N_{PE}^{max}$$

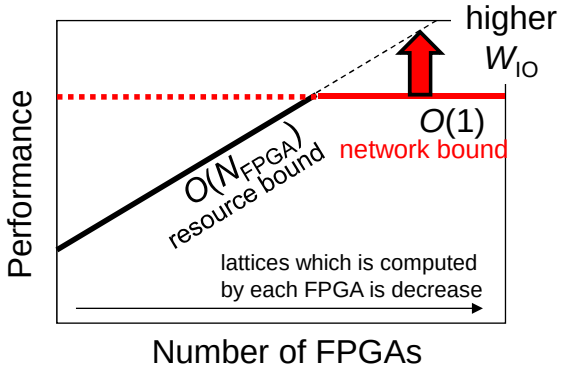
- For number of FPGAs (N_{FPGA}),

$$(\textit{Total Performance}) = O(N_{PE}^{max} \times N_{FPGA})$$

Performance Limited by Network-Bandwidth

- To utilize PEs for 100% cycles, communication time must be hidden with computing time.

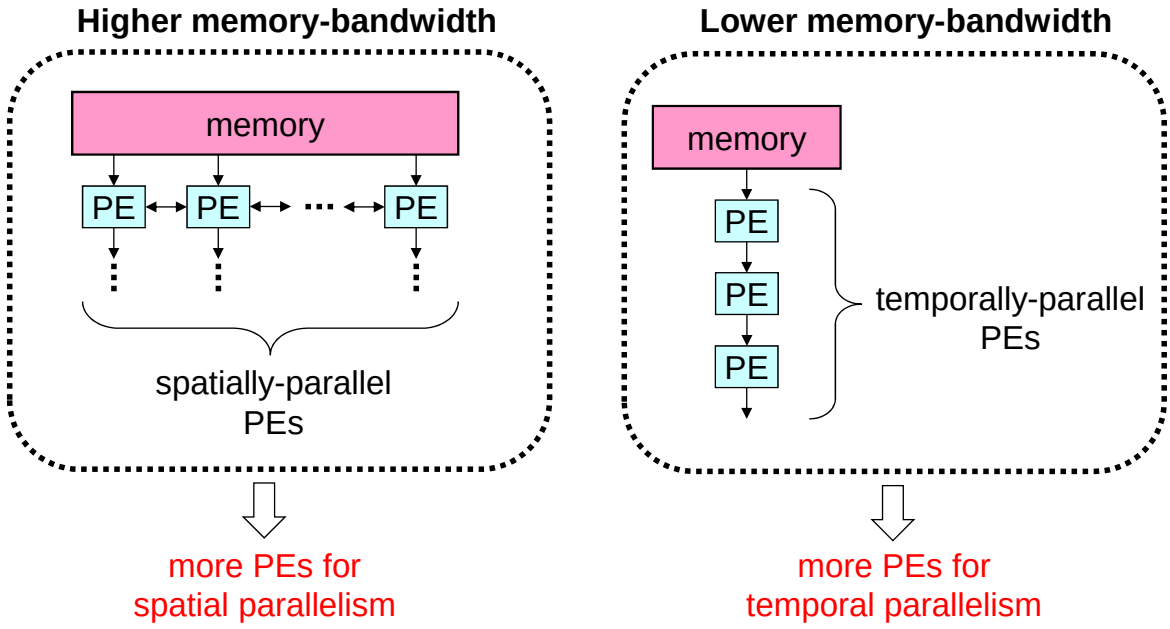




$$(\textit{comp. time}) = \frac{(\textit{lattice size})}{(\textit{performance})} \geqq (\textit{comm. time}) \left(\propto \frac{1}{W_{IO}} \right)$$

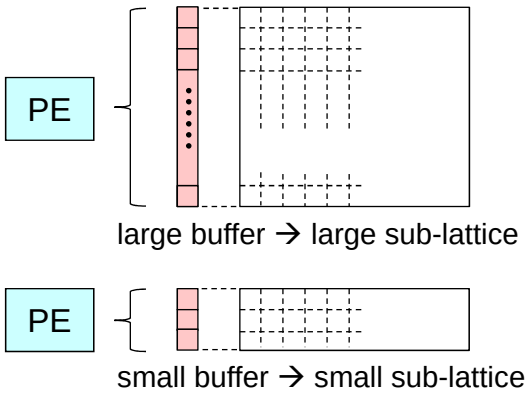
$$(\textit{performance}) \leqq \alpha(W_{IO})$$

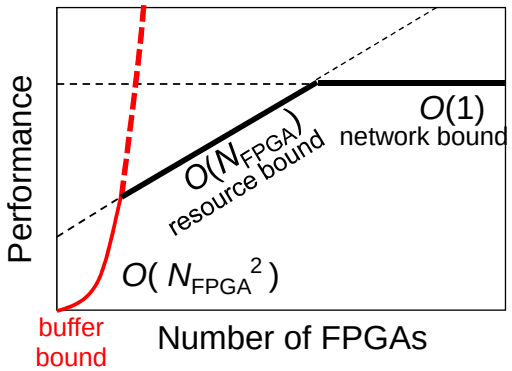
No Limitation by Memory-Bandwidth



Total performance is not limited by memory-bandwidth

Performance Limited by Buffer Size





- When using fewer FPGAs, a larger sub-lattice is allocated to each PE in strong-scaling.
- We must reduce N_{PE} so that each PE has a larger buffer.
- Less FPGAs, and less N_{PE} on FPGA

Performance ∝ # of PEs = O((Number of FPGAs)^2)

Parameters Used for Analysis

We designed LBM PEs for single-precision by using Flopoco [dinechin 2011].

Design / FPGAs	$N_{\mathrm{PE}}^{\max}$ 125MHz	B [MB]	W_{mem} [GB/s]	$W_{\mathrm{I/O}}$ [GB/s]
Stratix IV EP4SGX230 (DE4)	6	1.6	6	1
Stratix V 5SGXB6	6	6.6	24	116
Stratix V 5SGSD8	26	6.4	36	84

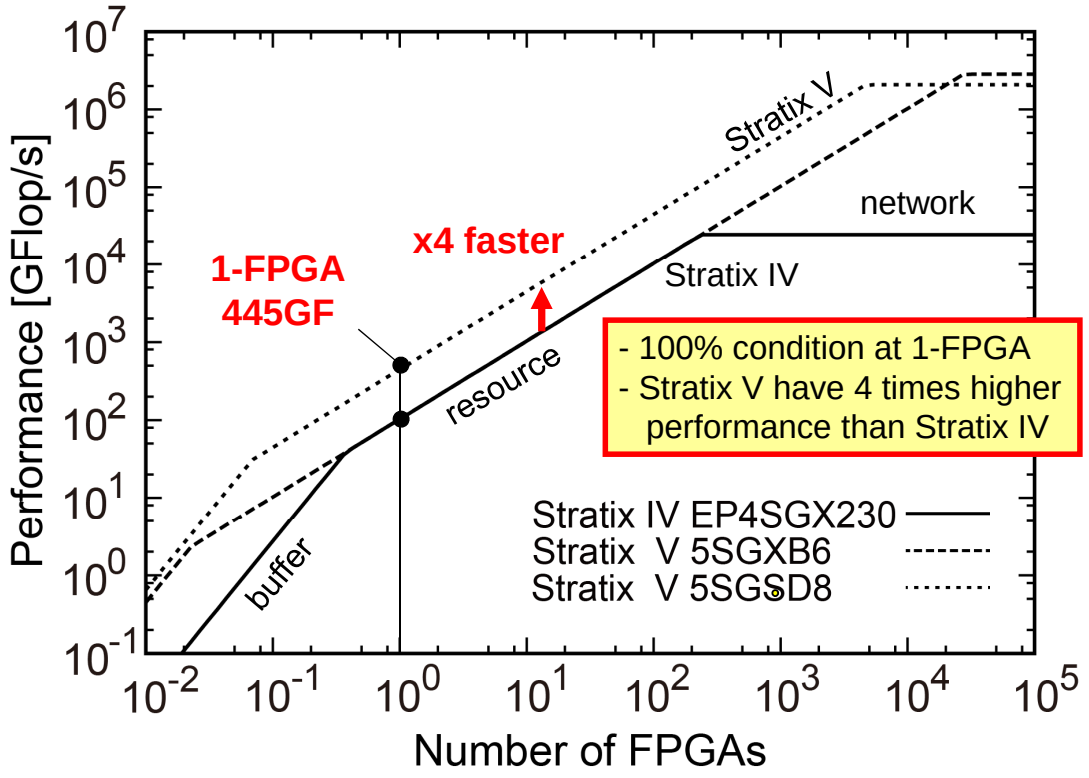
For Stratix IV

- ✓ estimate memory- and network- bandwidth available on DE4 Board

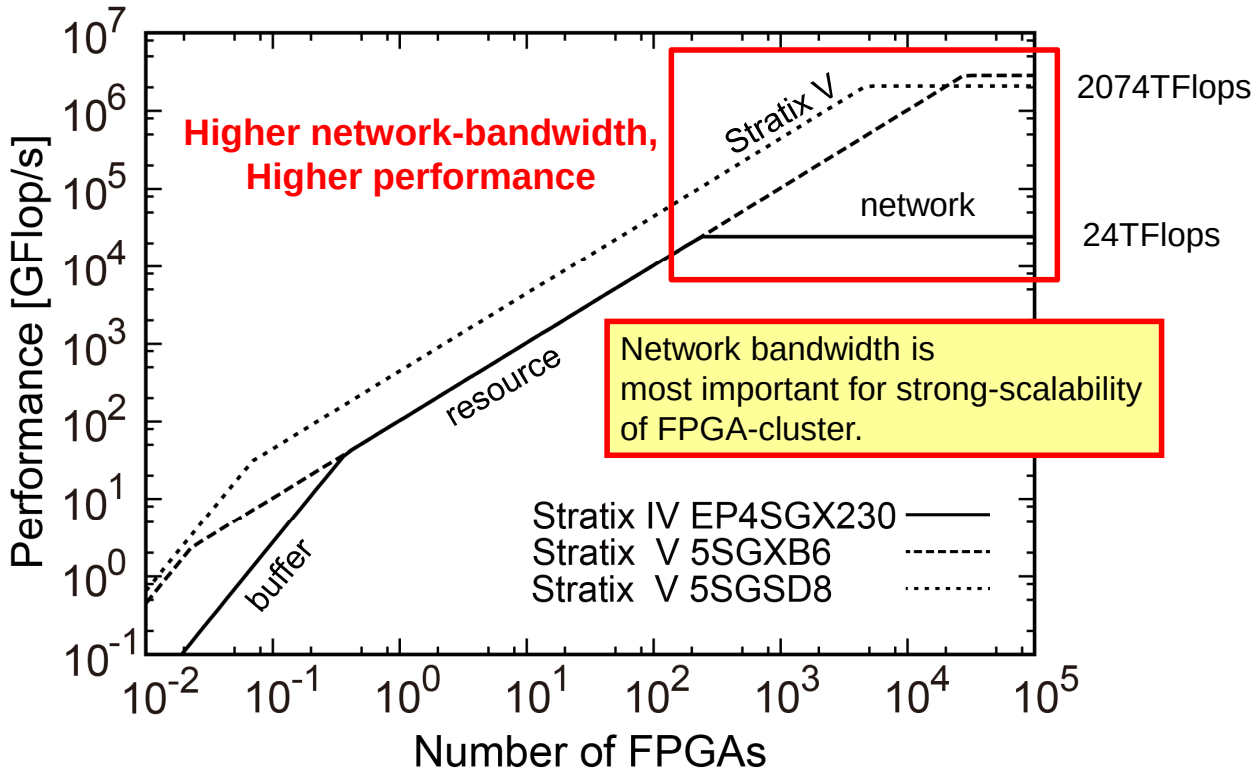
For Stratix V

- ✓ estimate parameters with a spec sheet

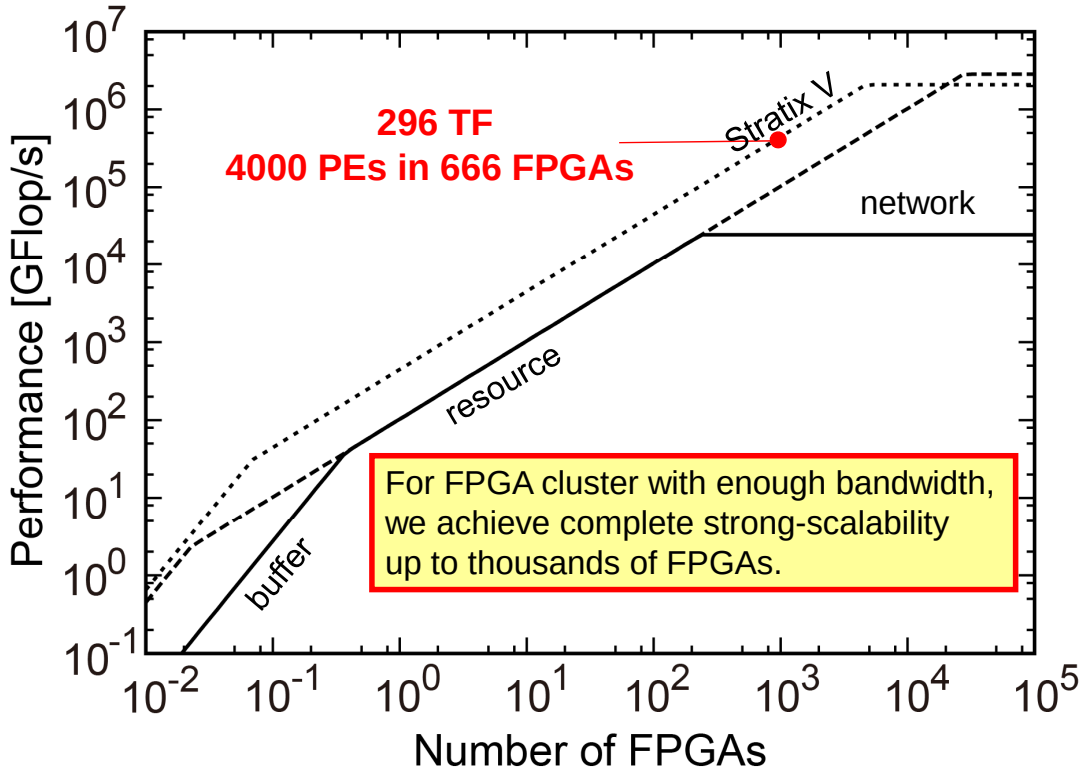
Strong-Scalability Analysis (4000×4000 Lattice)



Strong-Scalability Analysis (4000×4000 Lattice)



Strong-Scalability Analysis (4000×4000 Lattice)



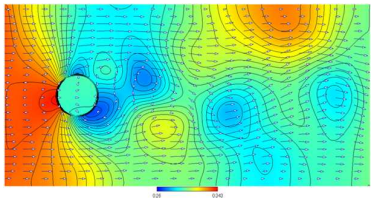
Conclusions

Strong-scalability analysis of tightly-coupled FPGA-cluster
Architecture design for scalable LBM on FPGA-cluster

- **Stream architecture for LBM computation**
  - Exploit spatial- and temporal-parallelism of stream-computation
  - Balance arithmetic-performance with bandwidth
- **Strong-scalability analysis for the tightly-coupled FPGA-cluster**
  - Formulate performance model and estimate performance of the cluster
  - Single Stratix V has 445GF
    (GeForce GTX-590 has peak performance of 2.488 TFLOPs)
  - Stratix V can scale up to 666 FPGAs, delivering 296TF

We can achieve good scalability
if FPGA cluster has ADN with sufficient bandwidth.

Future Work



- Implementation and benchmarking
- Performance model for 3D LBM computation
- FPGA Cluster with multi nodes of Stratix V FPGAs

DE5 was released!